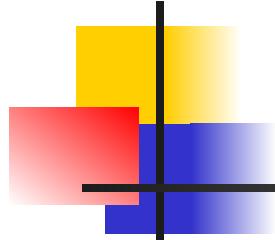


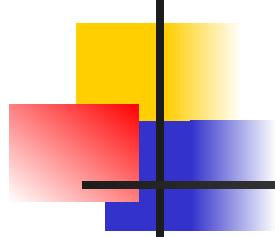
一、选择题（一）

- 1. 结构化程序设计所规定的三种基本结构是---C---。
 - A. 主程序、子程序、函数
 - B. 树形、网形、环形
 - C. 顺序、选择、循环
 - D. 输入、处理、输出
- 2. 下列关于C语言的叙述错误的是--A--
 - A. 对大小写不敏感
 - B. 不同类型的变量可以在一个表达式中
 - C. main函数可以写在程序文件的任何位置
 - D. 同一个运算符号在不同的场合可以有不同的含义
- 3. 以下合法的实型常数是--C--
 - A. .E4
 - B. 2.5E
 - C. 3.
 - D. E7



一、选择题 (二)

- 4. 表示关系 $x > y \geq z$, 则正确的C语言表达式为——C——。
A. $x > y >= z$ B. $(x > y) \& (y >= z)$
C. $(y < x) \&\& (y >= z)$ D. $(x > y) \text{and} (y >= z)$
- 5. 有语句 `scanf (" %d, %d", &a, &b)`, 要使变量a, b分别得23, 45, 则正确的输入形式为——B——。
A. 23 45 B. 23,45 C. 23;45 D. 2345
- 6. 已知 `int i=10;` 表达式 " $20 - 0 <= i <= 9$ " 的值是——B——。
A. 0 B. 1 C. 19 D. 20
- 7. 已知 `int a=15;` 执行语句 `a=a<<2` 以后, 变量a的值是——C——。
A. 20 B. 40 C. 60 D. 80



一、选择题 (三)

- 8. 对二维数组的正确说明是 C。
A. `int a[][]={1,2,3,4,5,6};` B. `int a[2][]={1,2,3,4,5,6};`
C. `int a[][3]={1,2,3,4,5,6};` D. `int a[2,3]={1,2,3,4,5,6};`
- 9. 若以下选项中的变量已正确定义，则正确的赋值语句是 C。
A. `x1=26.8%3` B. `1+2=x2` C. `x3=0x12` D. `x4=1+2=3`
- 10. 设有以下定义

```
#define d 2
int a=0;
double b=1.25;
char c= 'A' ;
```

则下面语句中错误的是 B/D。
A) `a++;` B) `b++` C) `c++;` D) `d++;`

一、选择题 (四)

- 11. 表达式 $5 \% 3 + 5 / 3$ 的值为 ---A---。
A. 3 B. 4 C. .666667 D. 4.666667
- 12. 下面C程序的输出是: --C--

```
main()
{ int m=5;
if( ++m>5)printf("%d\n",m);
else printf("%d\n",-m);
}
```

A. 4 B. 5 C. 6 D. 7

一、选择题 (五)

- 13. 有以下程序

```
main()
{
    char a,b,c,d;
    scanf( "%c,%c,%d,%d" ,&a,&b,&c,&d);
    printf( "%c,%c,%c,%c\n" ,a,b,c,d);
}
```

若运行时从键盘上输入： 6,5,65,66<回车>。 则输出结果是--A--
A. 6,5,A,B B. 6,5,65,66 C. 6,5,6,5 D. 6,5,6,6

- 14. 若变量已正确定义，要求程序段完成求5!的计算，不能完成此操作的程序段是--B---。

A. for(i=1,p=1;i<=5;i++) p*=i;
B. for(i=1;i<=5;i++) { p=1; p*=i; }
C. i=1;p=1;while(i<=5){p*=i; i++;}
D. i=1;p=1;do{p*=i; i++; }while(i<=5);

一、选择题 (六)

- 15. 有以下程序

```
main()
{
    int i=0,x=0;
    for (;;)
    {
        if(i==3||i==5)
            continue;
        if (i==6) break;
        i++;
        s+=i;
    }
    printf("%d\n",s);
}
```

程序运行后的输出结果是---D---。
A. 10 B. 13 C. 21 D. 程序进入死循环

二、问答题（一）

- 1. 有以下程序

```
main()
{
    char k;
    int i;
    for(i=1;i<3;i++)
    {
        scanf( "%c" ,&k);
        switch(k)
        {
            case '0' :printf( "another\n" );
            case '1' :printf( "number\n" );
        }
    }
}
```

another
number
number

程序运行时，从键盘输入：01<回车>，程序执行后的输出结果是：

二、问答题（二）

- 2. 以下程序的功能是什么？

```
main()
{
    int i,s=0;
    for(i=1;i<10;i+=2)
        s+=i+1;
    printf( "%d\n" ,s);
}
```

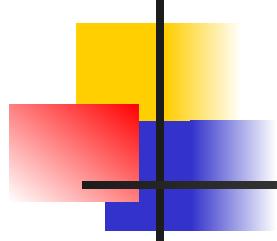
小于等于10的
偶数之和

二、问答题（三）

- 3. 以下程序的功能是什么？

```
#include <stdio.h>
main()
{
    int a,b;
    scanf("%d%d", &a, &b);
    printf("a=%d,b=%d\n", a, b);
    a= a + b ;
    b= a - b ;
    a= a - b ;
    printf("a=%d,b=%d\n", a, b);
}
```

交换两个整数

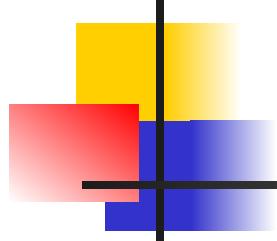


二、问答题（四）

- 4. 以下程序段的输出结果是

```
int a=10,b=50,c=30;  
if(a>b)  
    a=b;  
    b=c;  
    c=a;  
printf("a=%d b=%d c=%d\n",a,b,c);
```

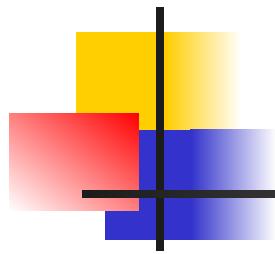
a=10 b=30 c=10



二、问答题（五）

- 若c已经正确定义，
`while((c = getchar()) != '\n');`语句的功能是
-----。

从键盘输入字符，直到输入为回车时停止。



三、 编程题（一）

- 1. 读取两个字符串，并将它们连接起来而不用到strcat函数(假设两个字符串str1, str2分别存储在两个长度为100的字符数组里，且str1的数组足够容纳str2的内容)。

```
char * strcat(char str1[], char str2[])
{
    int i = 0, j = 0;

    //move to the end character '\0' of str1;
    for(; str1[i] != '\0'; i++);

    //copy characters in str2 to str1 until '\0'
    while(str2[j] != '\0')
    {
        str1[i + j] = str2[j];
    }

    //end str1 with '\0'
    str1[i] = '\0';

    return str1;
}
```

```
#include <stdio.h>

void main()
{
    char str1[100], str2[100];
    gets(str1);
    gets(str2);
    strcat(str1, str2);
    printf("%s\n", str1);
}
```

三、编程题 (二)

- 2. 实现直接插入排序算法。
- 算法思想：将序列中的数据一个个插入。当插入第 i ($i \geq 1$) 个对象时，前面的 $V[0]$, $V[1]$, …, $V[i-1]$ 个数据已经排好序，这时，用 $V[i]$ 与 $V[i-1]$, $V[i-2]$, … 顺序进行比较，找到插入位置即将 $V[i]$ 插入，原来位置上的对象向后顺移。

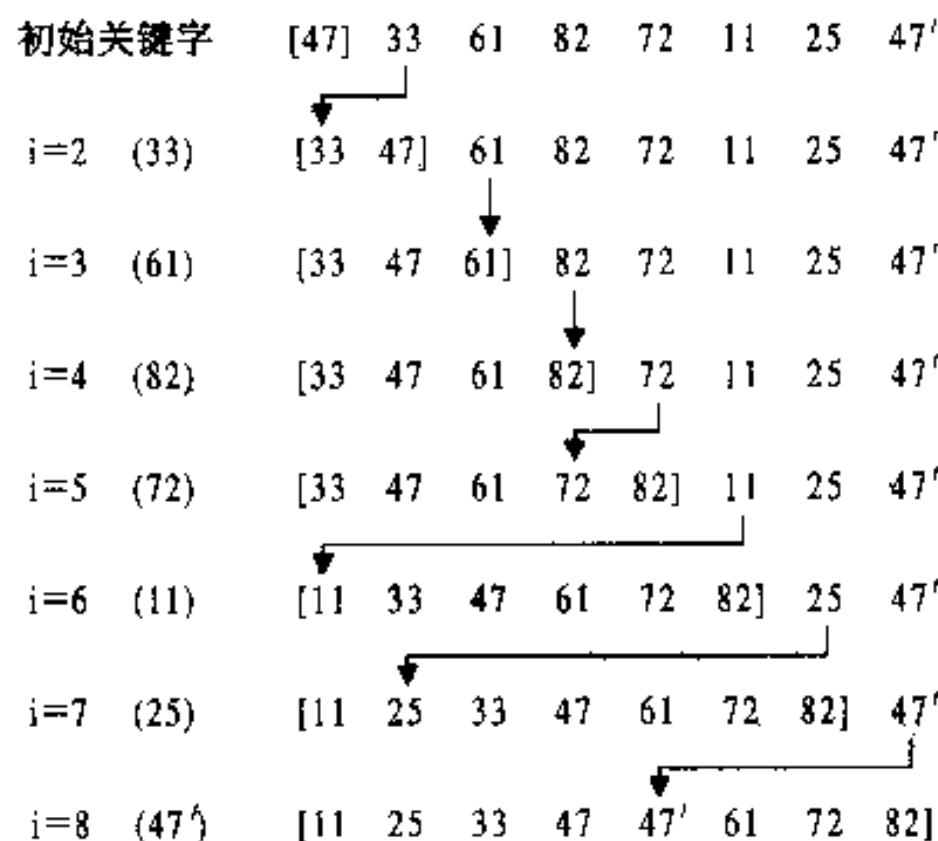


图 14.1 直接插入排序示例

```
void InsertionSort(int a[], int size)
{
    for(int i = 1; i < size; i++)
    {
        int temp = a[i];
        int j = i - 1;
        while(j >= 0 && temp < a[j])
        {
            a[j + 1] = a[j];
            j--;
        }
        a[j + 1] = temp;
    }
}
```

```
void main()
{
    int array[5]={8, 9, 28, 11, 23, 8};
    InsertionSort(array, 5);

    for(int i = 0; i < 5; i++)
    {
        printf("%d ", array[i]);
    }
    printf("\n");
}
```